CIDB
The PSI Controls Inventory DataBase

Timo Korhonen, PSI
(for the CIDB Team)

Introduction

- The Controls Hardware group at PSI is responsible of
  - Specifying (when we are lucky to be early enough…)
  - Purchasing and delivering to the users
  - Maintaining (repairs, upgrades)
  - Developing when needed

the control system hardware for three accelerators and the respective beamlines

more than 300 VME 64x crates, 150 CAMAC crates (to be gradually replaced as budget and manpower allows), more than 120 types of hardware (part types)

Introduction

- There is a lot of routine logistics involved
  - Maintain the stock level, order new parts
  - Send parts to repair (or repair in-house)
  - Deliver parts to the users, or install them
  - Keep track of the costs
- Expectations on quality
  - Hardware should be tested and tracked
  - Data on calibrations, firmware upgrades, faults and repairs should be available
  - Replacements (spares) have to always be available

CIDB is…

- A database-based application that
  - Enables us to track the control system hardware (stock, installations)
  - Embeds the working practices related to controls hardware distribution and maintenance (transparency)
  - Tries to make our (controls group HW&SW, system developers, other related groups) life easier
  - It is limited to the hardware handling, but is able to provide configuration information (installation hierarchy)
- Note: I am not a database expert; I will not show any table diagrams or othe details in this talk. However, details can be obtained from the developers if wanted.

Goals

- To know what hardware is installed where
  - Versions, (fault) history, firmware, pricing,…
- To know the stock status
  - To keep enough cards in stock, order early enough
- To improve quality
  - Introduce systematic tests, record results
- To ease the work
  - Our (hardware group) and the software developer's (requesting new hardware)

Goals

- Improve the way of working
  - Better defined work procedures
    - Define the ‘business rules’
  - Better tracking of what happened
    - All hardware moves are recorded (history), etc.
  - Easier requesting of hardware
    - Emails can only be seen by recipients, no overview of what has been requested and issued
  - Introduce user ‘roles’
    - Stock manager, system responsible

Some concepts

- Part
  - Component of the control system: VME crate, IOC, carrier board, ADC, DAC,…
- Connector
  - Defines how parts fit together (VME bus, IP bus, etc.)
- Health
  - Is the part known to be OK (tested), broken or unknown?
- System
  - A collection of parts to perform a control system task (typically, but not necessarily a VME crate with one IOC.)

The application

- Oracle database
- Web-based GUI (PHP)
- Developed in collaboration with the Northwest Switzerland Univ. of Applied Sciences (FHNW)

Workflow

- The control system hardware consists of **parts** of several types
- The parts are purchased and go to (one of a few) stock.
  - Before that, the parts undergo a functionality test (not fully implemented yet.)
- Users request (internal ‘order’) for hardware parts:
  - ‘I need a CPU card by next Monday’. Hopefully needs are known well in advance
- Parts are delivered from stock and installed in systems
- Faulty hardware comes back to the stock manager and is sent to repair. Replacement is requested
- Stock manager(s) are aware of what is in stock, what is requested, what is in repair, etc.
- History of each module (‘life history’) is recorded
- History of each system is recorded
- History of each request and delivery is recorded

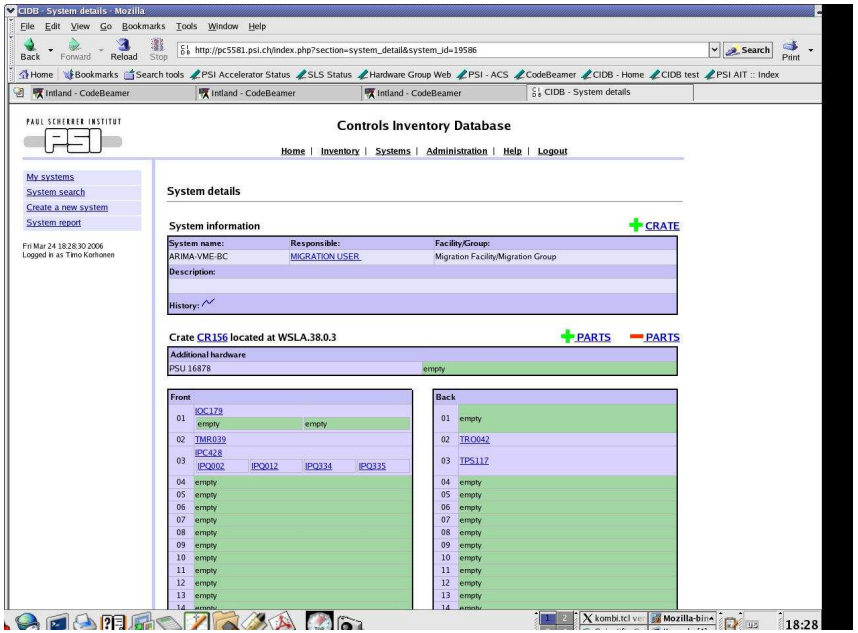
Installations

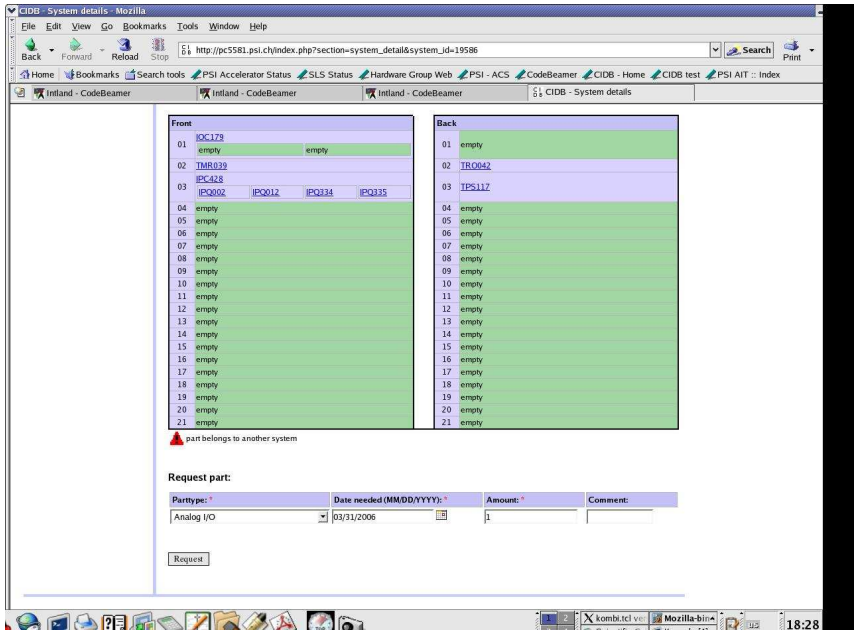
- A system is a collection of parts
  - Parts (usually) are hierarchically connected together (parent-child relationship)
  - Has a system responsible (and possible delegates)
- Each part is of a known part type
  - Part type defines the general properties (connectivity, type-specific information)
- Part connectivity is defined by a ‘connector compatibility matrix’

User roles

- Two user roles: Stock Manager, system user
- Stock manager can
  - Assign parts to users
  - Define new part types and their connectivity
  - Has full access to all information
- System user
  - Manages parts assigned to him/her
  - Requests new parts from the stock when needed
  - Has access to information of his/her systems
  - Read access for other systems

System view

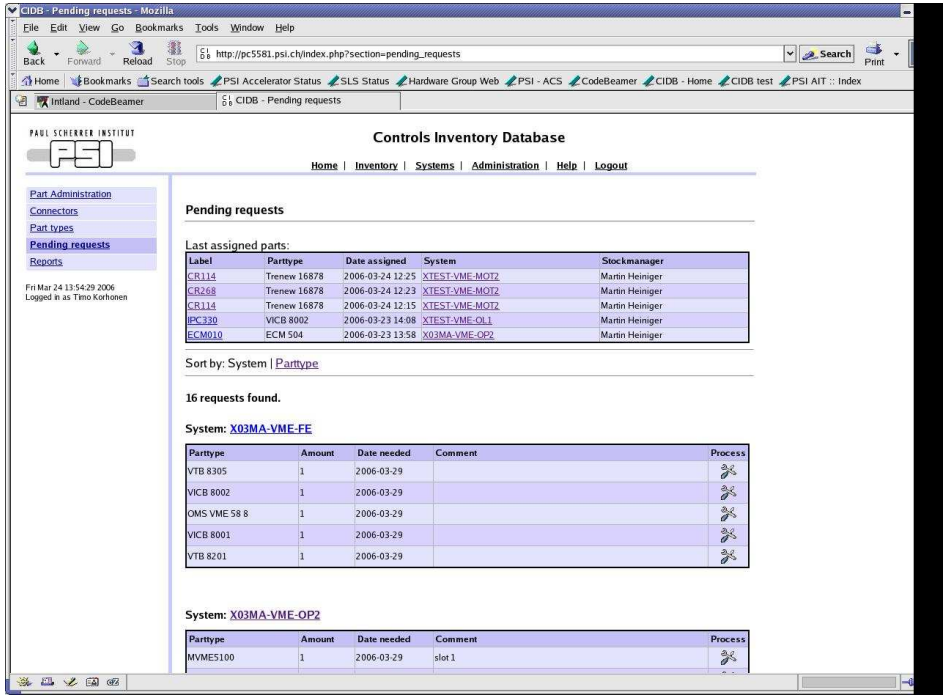




System contents (VME crate)
-nonhierarchical installations
are also supported

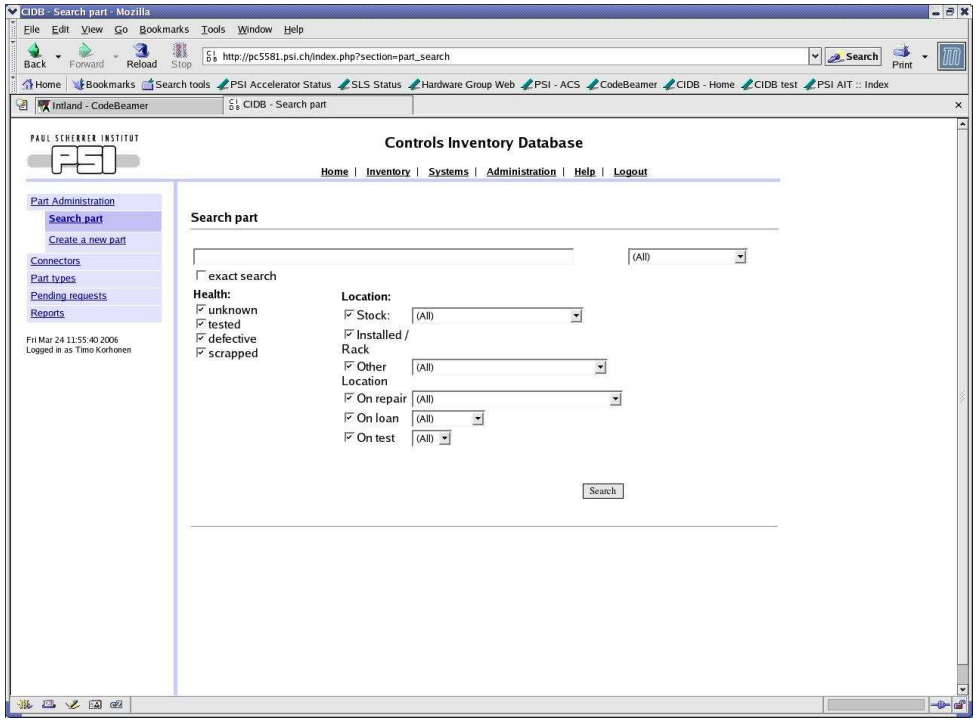
Request part:
-select part type from a list
-add count & date needed

Stock manager's view



Requested parts view:
-all hardware requests are listed (by system/parttype/date needed)
-stock manager selects the part from stock and puts it in

Part search

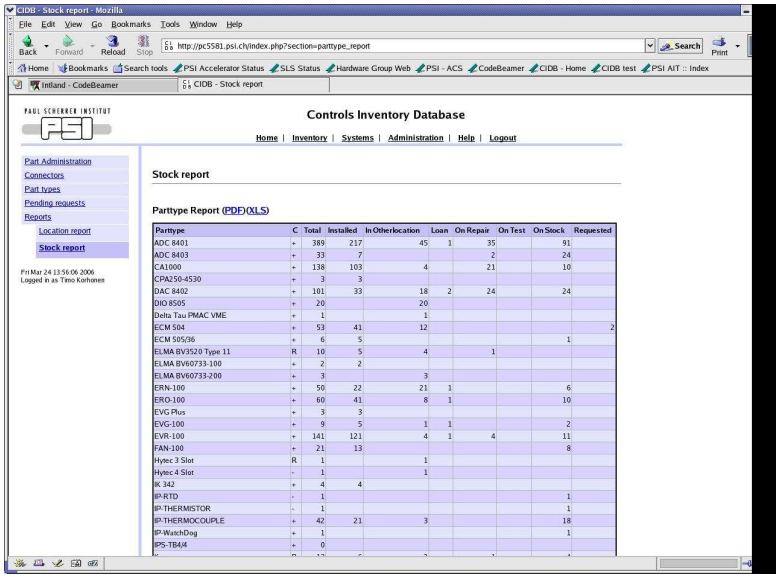


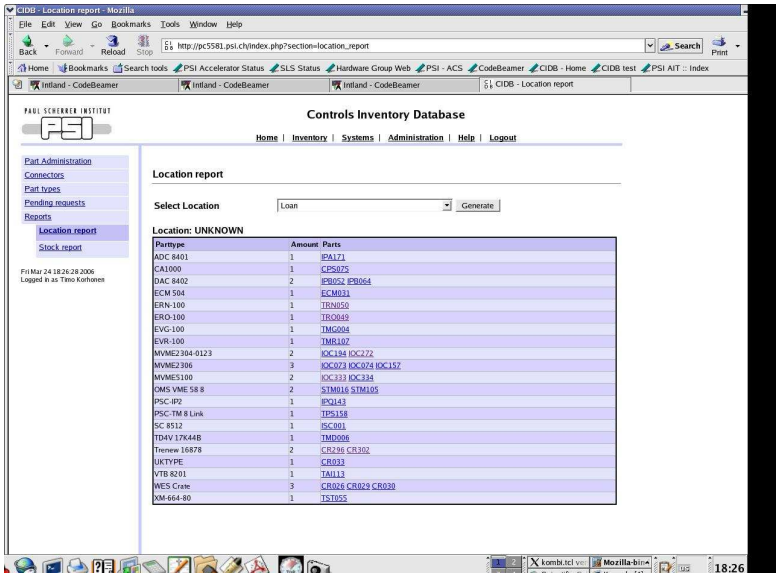
Each part (hardware card) has a unique ID
-can be searched by its ID, location, health (tested, defect, unknown,…)
-the part's full history is available (when purchased, installed, tested, moved, broken, repaired,…

Available to any user.

PAUL SCHERRER INSTITUT

Reports





Stock report
-what is available, installed, requested, etc.

Location report
-what is at a specific location (stock, rack,..)

More: system report (configuration, pricing, etc.)
-system history (who has changed what parts and when)
And many more…

Status & future

- Core functions
  - In use since February
  - CIDB has already much improved our view of the status, and enabled to share and distribute the work better
- Integrate hardware testing
  - Work in progress
- Include purchasing information
  - Important for ordering, repairs, budgeting, etc.
  - In progress, planned by end of June
- New applications/modules
  - (diagnostics) component calibration management
- Connection to / integration with IRMIS?
  - The functionality is largely complementary

What would be the gain?

-Integrate (routine) logistics to the control system lifecycle
-Configuration information would be automatically provided
-Possibility to improve quality management (systematic faults in some components, tracking of firmware/driver compatibility,…)
-follow the whole chain of signals from the source to the PV
-no need to record the same information twice

What is not clear to me:

-how much of this is already in IRMIS
-how much effort it would be to integrate?
-is it interesting for others than us?